

final-prj Hungry Snake

AP Class :
Ms. Bathaeian

Sajad Dehqan - 40012358014

Hungry_Snake

کلاس های استفاده شده:

App.hpp

Frog.hpp

Snake.hpp

Game.hpp

همانطور که از هدر فایل های hpp مشخص است فاز گرافیکی نسبت به فاز لاجیک یک کلاس game را نیز دارا است که بخش عملیات اجرای گرافیکی بازی متکی بر SFML بر دوش آن است.

Class app:

```
class App
{
private:
    Game game;
    void menu();

public:
    App();
};
```

این کلاس در راستای مدیریت هرچه بهتر بازی ساخته شده تا رویداد ها در آن کنترل و بررسی شود.

در قسمت public این کلاس یک کانستراکتور App() مشاهده میشود که هنگام ساخت شیء ای از اپ آن فراخوانی میشود و تابع MENU() را فراخوانی میکند، در این تابع اجرای منوی صورت گرافیکی صورت میگیرد.

Class frog:

```

class Frog
{
private:
    float frogSpeed = 1;
public:
    int frogStartpos();
    void frog_setSpeed();
    float frog_getSpeed();
    void frog_speed_reset();
};

```

قورباغه یک مختصات x به صورت رندوم دارد که از بالا به پایین بیاید و تولید این مختصات بر عهده تابع frogStartpos() می باشد.

قورباغه ها پس از بارش پس از مدت معینی سرعتشان بیشتر می شود که این افزایش سرعت بر عهده frog_setspeed() می باشد و پس از باخت کاربر سرعت باید توسط frog_speed_reset() به مقدار اولیه خودش برگردد.

Class snake:

```

class Snake
{
private:
    int snakeSpeed = 10;
public:
    void set_snakeSpeed();
    void reset_snakeSpeed();
    int get_snakeSpeed();
};

```

در راستای افزایش سرعت قورباغه ها لازم است که سرعت حرکت مار توسط کاربر هم افزایش داشته باشد که بدین صورت افزایش می یابد و پس از شروع مجدد بازی سرعت باز نشانی میشود.

Class game:

```
class Game
{
private:
    int score = 0;
    std::string set_score();
    void tryAgain(sf::RenderWindow &window, sf::Text point);
    //----- set sound -----
    sf::SoundBuffer back_sound;
    sf::SoundBuffer eat_sound;
    sf::SoundBuffer gameover_sound;
    //-----
    //----- set font -----
    sf::Font font;
    //-----

    //----- set photo -----
    sf::Texture texture;
    sf::Texture snake_image;
    sf::Texture frog_image;
    sf::Texture frog2_image;
    //-----

public:
    // Game(/* args */);
    Frog frog1;
    Frog frog2;
    Snake snake;
    void runGame(sf::RenderWindow &window);
};
```

اما یکی از مهمترین کلاس هایی که وجود دارد گیم است که در تابع runGame() آن سه بخش صورت میگیرد.

1_ بررسی ورودی های داده شده توسط کاربر

2_ بررسی منطق و قوانین بازی

3_ رندر اسکرین

تصاویر و صداهاى مربوطه بازی را این کلاس شامل خود دارد و دو شی از کلاس قورباغه برای بارش آنها و یک کلاس از مار جهت کنترل آن.

Main:

```
int main()
{
    App app;

    return 0;
}
```

در main برنامه یک شیء از App ساخته میشود که کانستراکتور آن متد منو و سپس استارت بازی را فرا میخواند.

نمونه ای از اجرای برنامه در فایل game_play قرار داده شده است.

Git address:

https://github.com/sedwna/Hungry_Snake